

Curso de Python

1. Función Range

La función range nos sirve para crear en forma rápida un objeto iterable.

```
número a partir del cual inicia el rango (incluido)

range(valor_inicio, valor_final, paso)

número antes del cual el rango finaliza (no incluido)

print(list(range(1,13,3)))
>> [1,4,7,10]
```

Ejemplo: supone que necesitamos una lista que va del 1 al 100.

```
lista * list(range(1,101))
print(lista)
```

2. Función Enumerate

La función enumerate() nos facilita llevar la cuenta de las iteraciones, a través de un contador de índices de un iterable, que se puede utilizar de manera directa en un loop, o convertirse en una lista de tuplas con el método list().



Ejercicio 19: realice este programa e indique que es lo que se muestre en pantalla como resultado.

```
lista = ['a','b','c']
indice = 0

for item in lista:
   print(indice,item)
indice += 1
```

Ejercicio 20: realice este programa e indique que es lo que se muestre en pantalla como resultado.

```
for item in enumerate(lista):
    print(item)
```



3. Función random

En realidad random es un módulo que tiene varias funciones que nos permiten generar números aleatorios de distintos tipos.

```
Nombre del módulo

from random import * = Todos los métodos

También pueden importarse de manera
independiente aquellos a utilizar.

randint(min, max): devuelve un integer entre dos valores dados (ambos límites
incluidos)

uniform(min, max): devuelve un float entre un valor mínimo y uno máximo

random(sin parámetros): devuelve un float entre 0 y 1

choice(secuencia): devuelve un elemento al azar de una secuencia de valores
(listas, tuples, rangos, etc.)

shuffle(secuencia): toma una secuencia de valores mutable (como una lista), y la
retorna cambiando el orden de sus elementos aleatoriamente.
```

4. Función Min y Max

Las funciones Min y Max nos devuelven el valor mínimo o máximo dentro de un objeto.

```
ciudades_habitantes = {"Tijuana":1810645, "Leóh":1579803}
lista_valores = [5**5, 12**2, 3050, 475*2]

print(min(ciudades_habitantes.keys()))
>> León

print(max(ciudades_habitantes.values()))
>> 1810645

print(max(lista_valores))
>> 3125
```